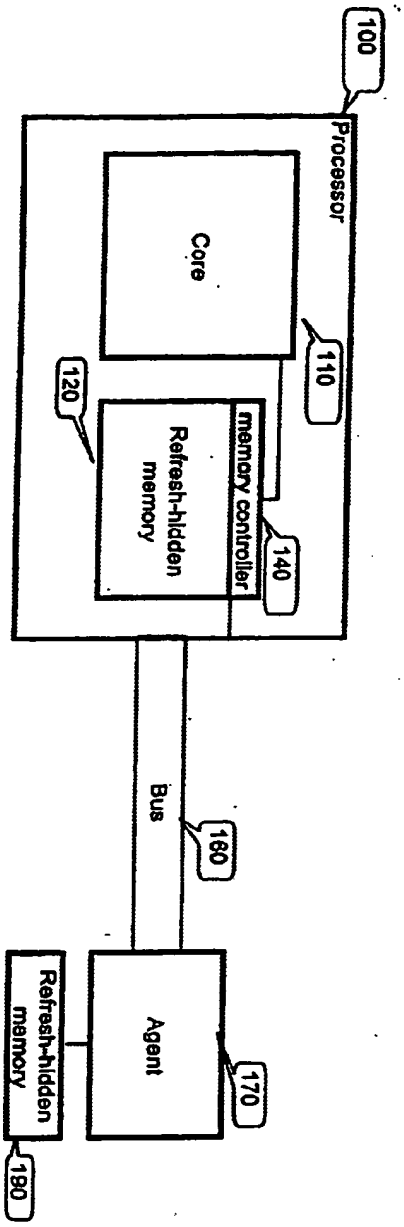


Figure 1

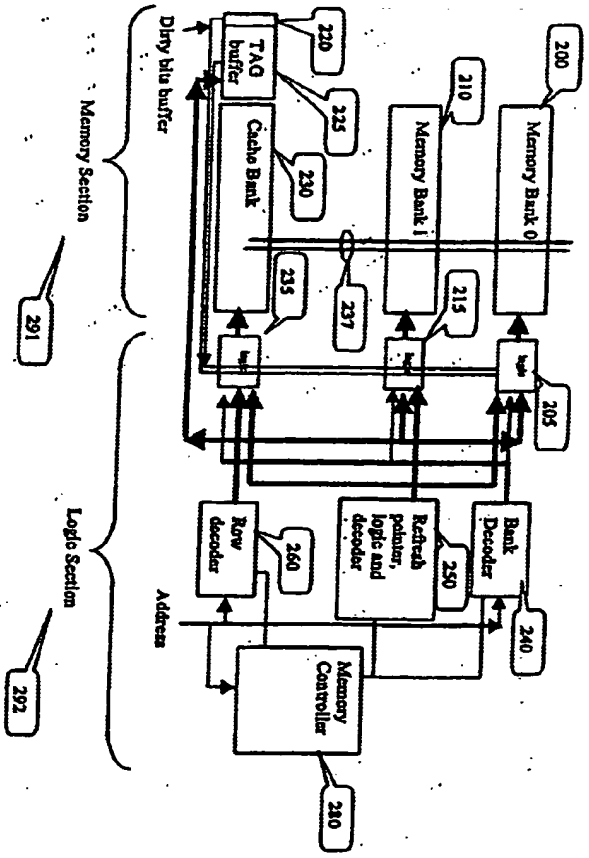


Figure 2

Figure 3

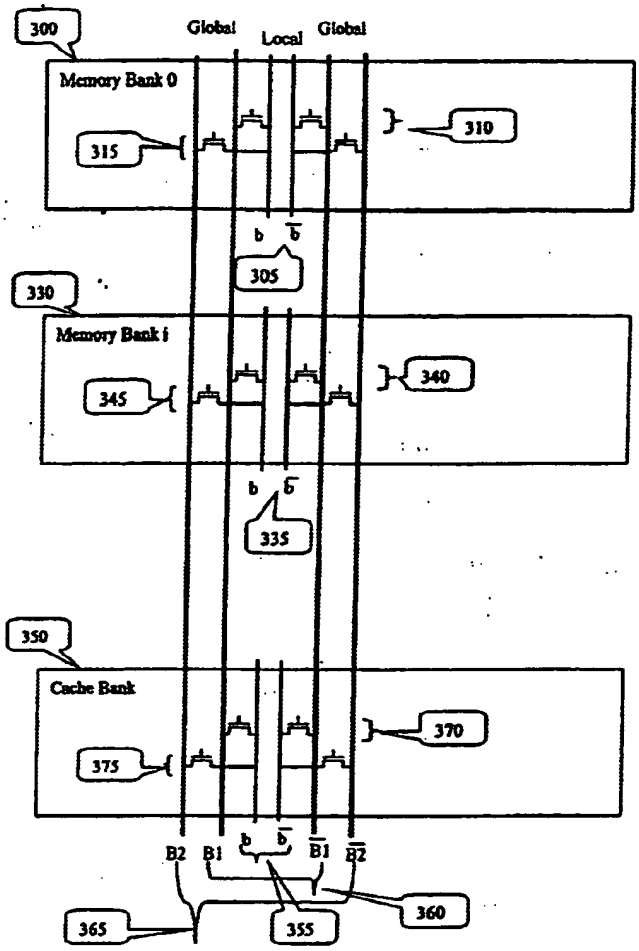
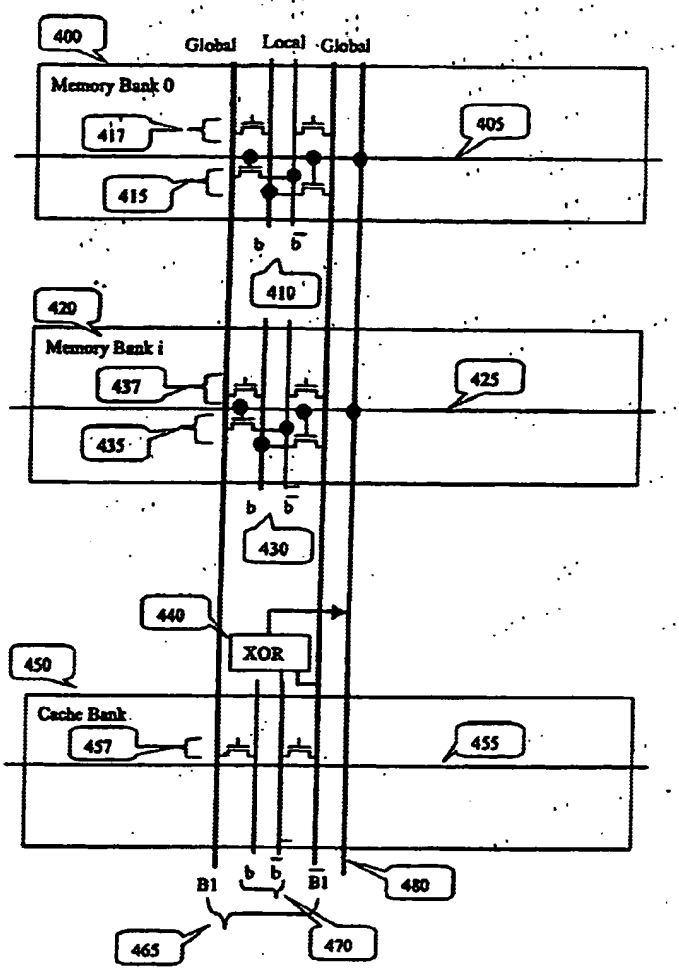


Figure 4



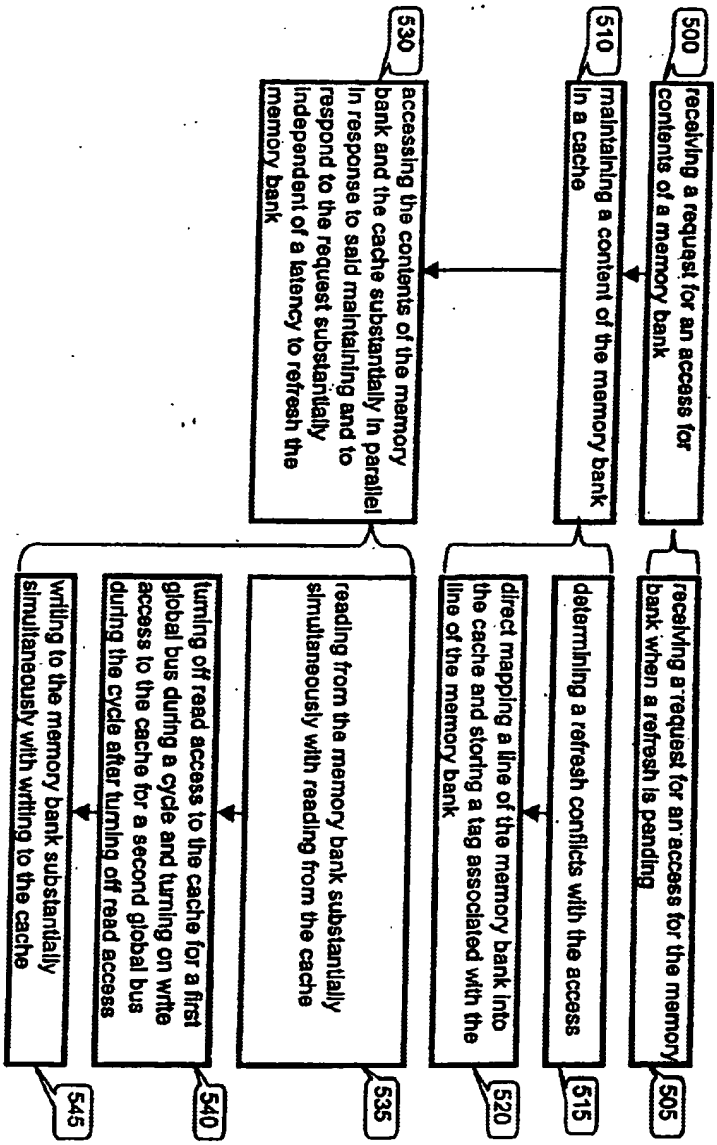


Figure 5

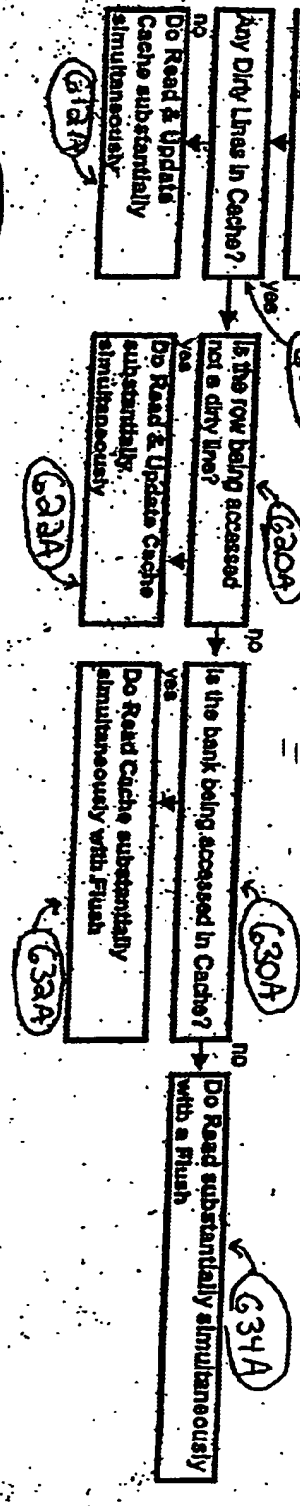


Figure 6a

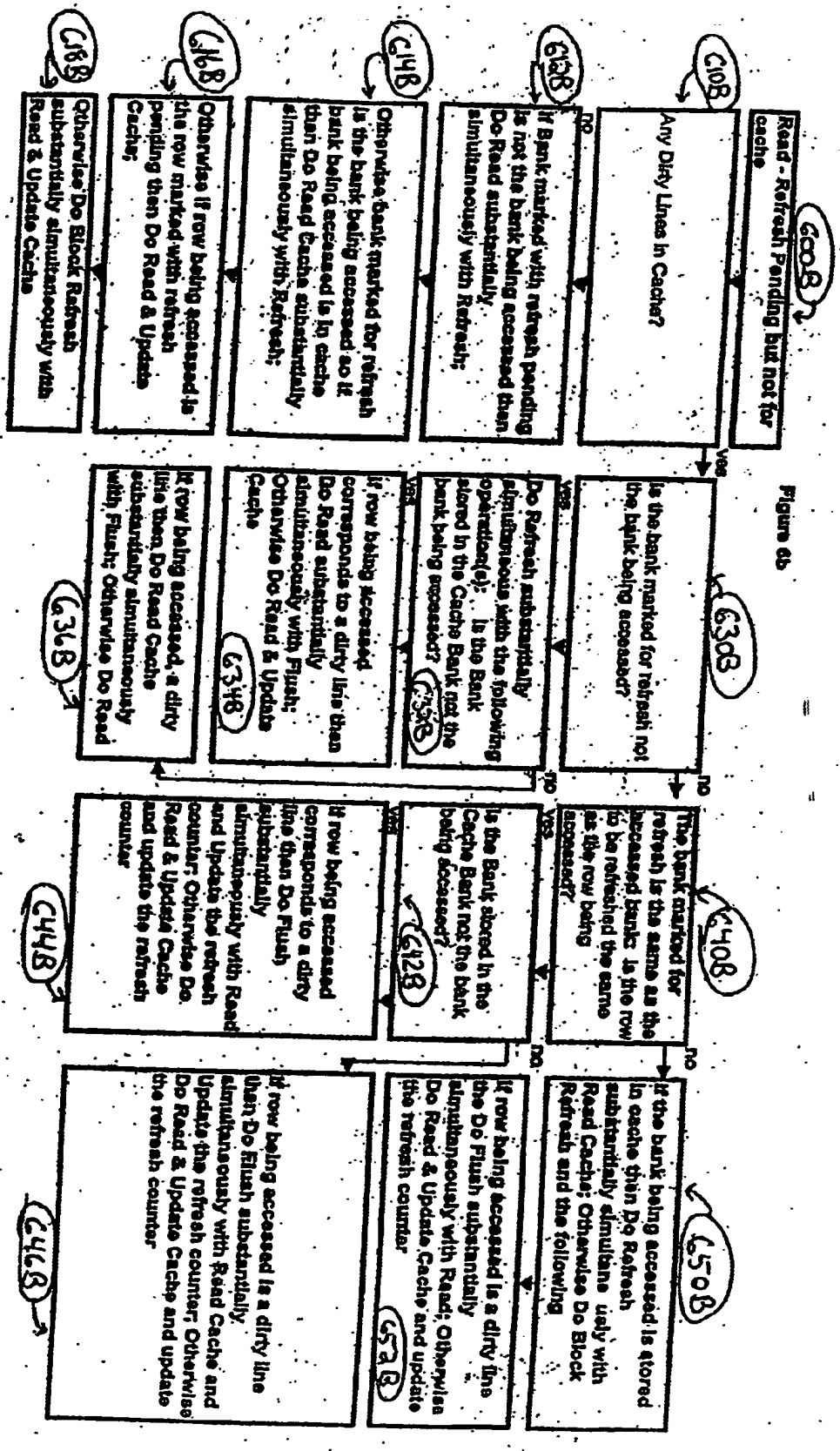


Figure 6b

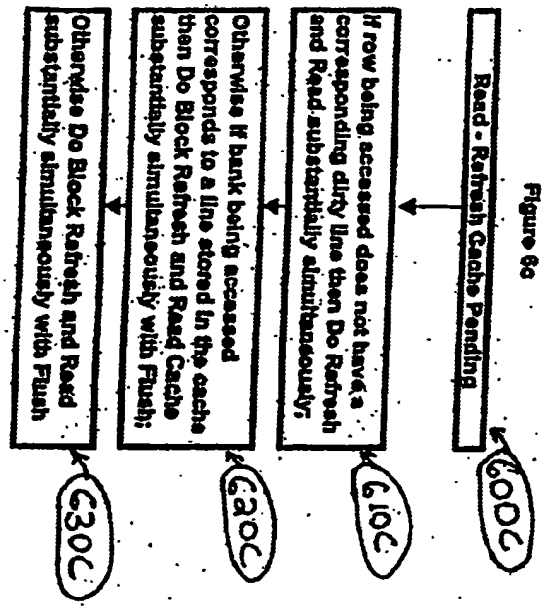


Figure 6c

Figure 6d

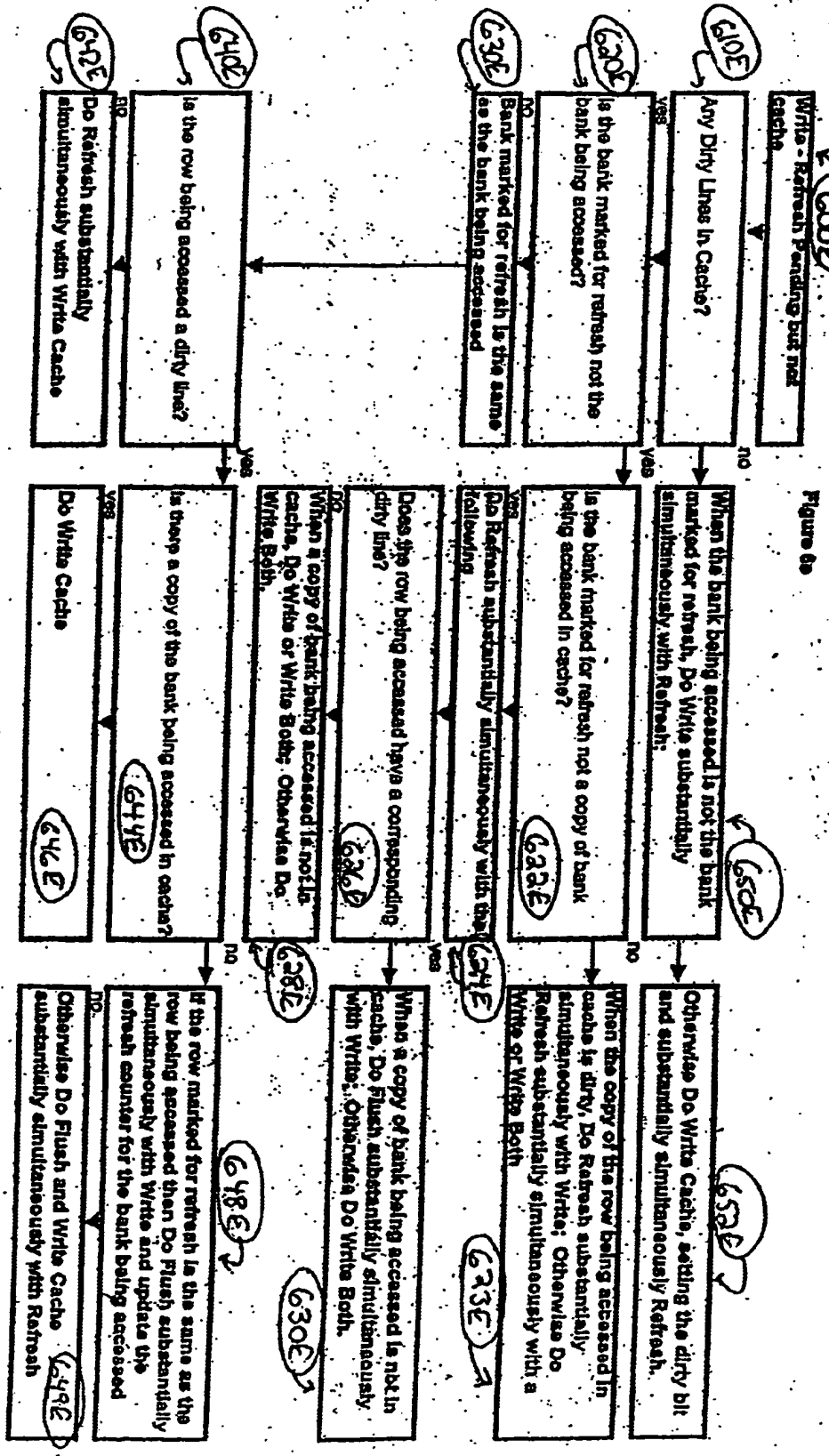


Figure 6a

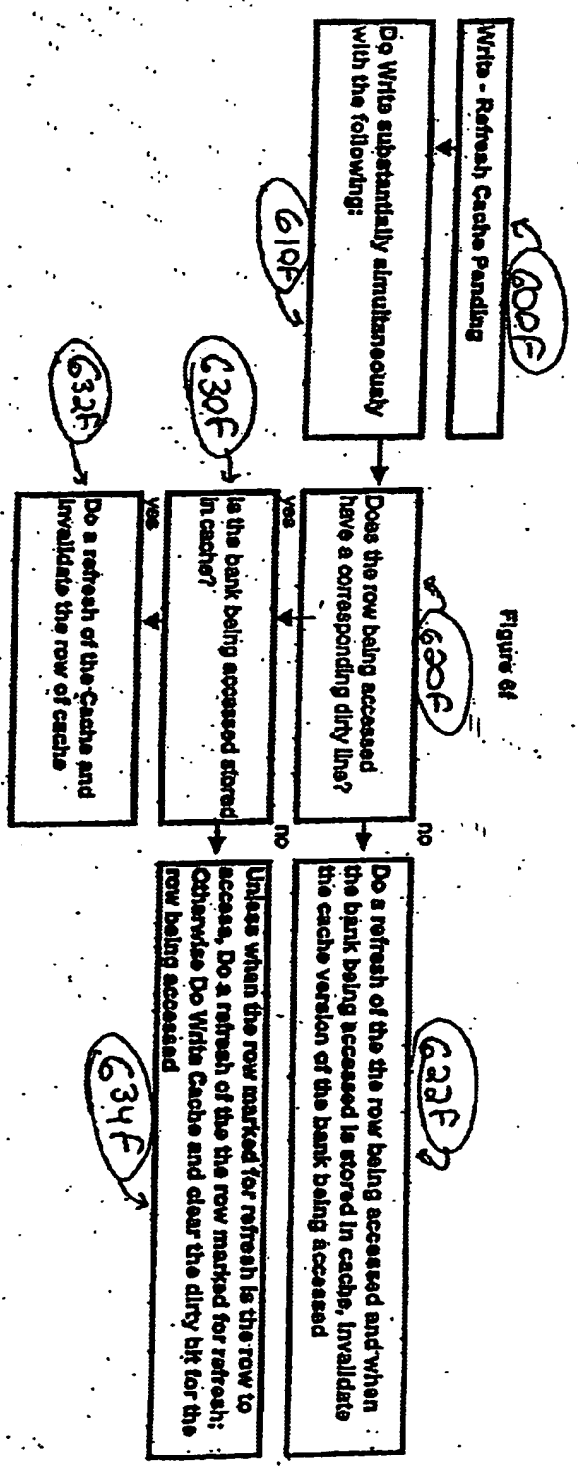


Figure 6f

Figure 7A

Inputs

```
Refresh Counter:          rb, rr    /* refresh bank, refresh row */
Current Access Address:   wb, wr    /* access (write) bank and row */
Number of rows in a bank: row
```

Buffers

```
Valid[row]          /* valid bits for each of the rows in the cache bank */
Dirty[row]          /* dirty bits for cache lines*/
Cache[row]          /* cache bank */
Cachetag[row]       /* Tag for cache lines */
```

Functions

```
Refresh()     {mem[rb, rr] = mem[rb, rr]; }
Flush()       {mem[cachetag[wr], wr] = cache[wr] $$ dirty[wr] = 0;} /* WB dirty line */
ReadUpdateC() {data_out = mem[wb, wr] $$ cache[wr] = mem[wb, wr]
                    $$ cachetag[wr] = wb; }
Read()        {data_out = mem[wb, wr]; }
Write()       {mem[wb, wr] = data_in; }
ReadCache()   {data_out = cache[wr] $$ cache[wr] = cache[wr];}
WriteCache()  {cache[wr] = data_in $$ cachetag[wr] = wb $$ dirty[wr] = 1;}
WriteBoth()   {Write() $$ Writecache()$$ dirty[wr] = 0; }
              /* write to both the cache and memory */
```

($$ - means operations are done in parallel)

pseudo program description

```
/* Initialize all cachetags to be NULL */
Cachetag[*] = NULL;

/*** READ ***/

/*--------- No  refresh pending ---------- */
if (no refresh pending) {
        if (no dirty line) {ReadUpdateC(); }
        else /* there is a dirty line */ {
                if (!dirty[wr]) { ReadUpdateC();}
                else if (wb == cachetag[wr]) { ReadCache() $$ Flush() $$ dirty[wr] = 0; }
```

```
/* ---------Refresh pending ---------*/
else {
        If (rb != cache) {
                If (no dirty line) {
                        if (rb != wb) { ReadO $$ RefreshO; }
                        else { /* rb == wb */
                                if (wb == tag[wr])
                                        { ReadCacheO $$ RefreshO; }
                        else {
                                if (wr == rr) ReadUpdateCO;
                                else Block_refresh $$ ReadUpdateCO;
                        }
                }
        else /* there is a dirty line */
                if (rb != wb) {
                        RefreshO $$
                        if (tag[wr] != wb) {
                                if (dirty[wr]) { FlushO $$ ReadO;}
                                else ReadUpdateCO;
                        }
                        else {
                                if (dirty[wr]) { ReadCacheO $$ FlushO;}
                                else ReadO;
                        }
                } /* end if rb != wb */
                else { /* refresh bank is the same as the accessed bank */
                        if ( wr == rr) {
                                if (tag[wr] != wb) {
                                        if (dirty[wr]) {FlushO $$ ReadO
                                                $$ UpdateRefreshCounterO; }
                                        else {ReadUpdateCO $$ Update Refresh CntrO;}
                                }
                                else {
                                        if (dirty[wr]) { ReadCacheO $$ FlushO
                                                $$ UpdaterefreshCntr;}
                                        else { ReadUpdateCO $$ UpdateRefreshCounterO;}
                                }
                        }
                        else {
                                if (tag[wr] == wb) { RefreshO $$ ReadCacheO;}
                                /* dirty status for the line unchanged */
                                else {
                                        BlockRefresh;
                                        if (dirty[wr]) { FlushO $$ ReadO; }
                                        else ReadUpdateCO;
                                }
                        }
                }
        }
}
```

Figure 7B

```
else /* the cache bank is being refreshed */ {
        if (dirty[wr]) {
            if (tag[wr] == wb) {
                BlockRefresh;
                ReadCache() $$ Flush();
            }
            else {
                BlockRefresh;
                Read() $$ Flush();
            }
        }
        else {
            Refresh()
            Read()
        }
    }
} /* end refresh pending */
```

Figure 7C

```
/*** WRITE ***/

/* ------- No pending refresh ------- */
if (no refresh pending) {
        /* there is no dirty line */
        if (no dirty line) {Write(); }
        else {    /* there is a dirty line */
                if (!dirty[wr])  { Write();}
                else {
                        if (Cachetag[wr] == wb) { WriteBoth() $$ dirty[wr] = 0; }
                        else {Write() $$ Flush();}
                }
        } /* end dirty line */
}
```

Figure 7D

```
/* ------ Pending refresh ------- */
else {
    if (rb != cache) {
        if (no dirty line) {
            if (wb != rb) { Write() $$ Refresh() }
            else { WriteCache() $$ dirty[wr] = 1 $$ Refresh() ;}
        }
        else {   /* there is a dirty line */
            if (rb != wb) { /* refresh bank is different from the access bank */
                if (rb != cachetag[wr]) {
                    Refresh() $$
                    If (dirty[wr]) {
                        If (cachetag[wr] != wb) {Flush() $$ Write();}
                        Else {Writeboth();}
                    }
                    else {
                        If (cachetag[wr] != wb) { Write(); } /* or writeboth() */
                        Else {Writeboth();}
                    }
                }
                else {
                    if (dirty[wr]) { Refresh() $$ Write();} /* cache line remains dirty */
                    else { Refresh() $$ Write() ;} /* Or writeboth() */
                }
            }
            else { /* refresh bank is the same as the access bank */
                If (dirty[wr]) {
                    If (cachetag[wr] == wb) { Writecache();}
                    else {
                        if (wr == rr) {/* lucky day*/
                            Flush() $$ write() $$ Updaterefreshcounter;
                        }
                        else {
                            Flush() and write to cache with new data ;
                            $$ Refresh();
                        }
                    }
                }
                else { /* the line is not dirty */
                /* create another dirty line */
                    Refresh $$
                    WriteCache();
                }
```

Figure 7E

```
else { /* cache bank is being refreshed */
        Write() $$
        if (dirty[wr]) {
                if (cachetag[wr] == wb) {
                        Refresh() $$
                        Cachetag[wr] = NULL;
                }
                else {
                        Refresh();
                        /* unless wr = rr then we writeback and clear dirty bit */
                }
        }
        else { /* not dirty */
                Refresh();
                if (cachetag[wr] == wb) { Cachetag[wr] = NULL;}
        }
}
```

Figure 7F

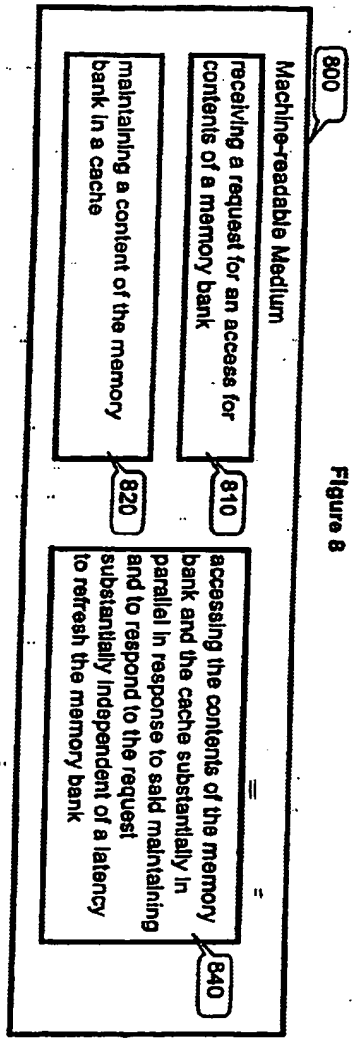


Figure 8